

Categorical Neural Abiogenesis: Policy-Indexed Quotients, Functorial Compression, and Certified Memory in No-Corpus Artificial Intelligence

Douglas C. Youvan doug@youvan.com www.youvan.ai

July 12, 2026

Artificial intelligence is normally described as learning representations of objects already defined by a training corpus, task ontology, or human labeling scheme. This paper develops an alternative formalism in which internal objects arise through policy-governed equivalence among generated states. A learning system is modeled by a category of internal presentations and admissible transformations together with a family of observation functors indexed by policies. Two presentations are identified when they are behaviorally substitutable under the relevant policy. Their equivalence classes form quotient objects, and compression becomes a functor that removes distinctions while preserving policy-relevant behavior. Training is represented as a developmental sequence of categories in which quotient classes may appear, stabilize, split, merge, and become reusable generators. Certified memory stores quotient objects together with policies, witnesses, countermodels, and canonical representatives rather than storing unqualified presentations. A policy-indexed HoTT interpretation treats equivalence certificates as paths and coherence among certificates as higher identity structure. The resulting framework extends Algebraic Abiogenesis from finite operations and term-function collisions to neural representations, causal circuits, and no-corpus artificial intelligence. It yields a precise Neural Abiogenesis Conjecture and a falsifiable experimental program for detecting the birth of internal objects during training and compression.

Keywords: Neural Abiogenesis, Algebraic Abiogenesis, no-corpus artificial intelligence, category theory, homotopy type theory, policy-indexed equivalence, quotient objects, model compression, certified memory, functional substitutability, mechanistic interpretability, representation learning

1. Introduction

Conventional artificial intelligence begins with preexisting objects. Corpora supply words, documents, labels, images, categories, relations, or demonstrations. Architectures then learn representations over this inherited ontology. Even unsupervised systems begin from data whose segmentation and observational units have already been chosen.

Algebraic Abiogenesis reverses this order. It begins with finite carriers, anonymous operations, formal composition, and equality of induced behavior. Stable term-function collisions generate provisional identities. Cross-order recurrence, explanation subtraction, countermodels, and

theory peeling separate coherent laws from finite accidents. The surviving identities define fully invariant congruences, quotient term algebras, canonical structures, and decision procedures.

The same methodology can be transferred to artificial intelligence. Neural states, activation patterns, computational traces, and circuits need not be regarded as intrinsic objects. They may be presentations that become object-like only when they are behaviorally substitutable under a specified policy. Training generates the presentations. Observation identifies relevant equivalences. Compression removes dispensable distinctions. Reuse promotes stable classes into higher-order generators.

The resulting hypothesis is that neural abstraction is a recursive quotient process. Internal objects are not primitive representational atoms. They are stable equivalence classes of generated states whose members may replace one another without violating policy-relevant behavior.

2. Generative Categories

Definition 2.1. Generative category

A generative category $\mathcal{G}$ consists of internal presentations as objects and admissible transformations as morphisms.

Objects may include:

activation states;

latent codes;

computational traces;

features;

subcircuits;

compressed representations;

previously certified internal objects.

A morphism $f: X \rightarrow Y$ is an admissible transformation between presentations. Examples include neural layers, attention operations, recurrent transitions, interventions, substitutions, compression maps, and compositions of learned modules.

Composition represents sequential computation. Identity morphisms represent unchanged presentations.

The category $\mathcal{G}$ is presentation-rich. Distinct objects may differ numerically, syntactically, architecturally, or developmentally while supporting the same functional behavior.

Definition 2.2. Generated subcategory

Given primitive objects P and generator morphisms F , the generated subcategory $\langle P, F \rangle$ is the smallest subcategory containing P and closed under identities and finite composition of morphisms in F .

This construction is the neural analogue of a term algebra. Primitive activations and transformations generate increasingly complex computational presentations.

Remark 2.3

The objects of $\mathcal{G}$ are not assumed to correspond to human concepts. They are implementation-level presentations. Conceptual objecthood appears only after an observational quotient is imposed.

3. Policy-Indexed Observation

Definition 3.1. Policy

A policy π specifies:

the contexts in which behavior is evaluated;

the downstream outputs to be preserved;

the admissible interventions;

the tolerated behavioral deviation;

the transformations under which substitutability is required;

the evidentiary threshold for certification.

Policies may be ordered by refinement. Write $\pi \leq \rho$ when ρ observes every distinction observed by π and possibly additional distinctions.

A refined policy induces a finer equivalence relation.

Definition 3.2. Observation functor

For each policy π , an observation functor is a functor

$$O\pi: \mathcal{G} \rightarrow \mathcal{B}\pi,$$

where $\mathcal{B}\pi$ is a category of policy-relevant behaviors.

$O\pi$ maps an internal presentation to its behavioral signature and maps an admissible transformation to its observable effect.

Behavioral signatures may include output distributions, task decisions, causal intervention responses, transition behavior, proof consequences, truth vectors, compression survival, or reuse profiles.

Definition 3.3. Policy equivalence

For $X, Y \in \text{Ob}(\mathcal{G})$, define

$$X \simeq_{\pi} Y$$

when $O\pi(X)$ and $O\pi(Y)$ are equal or equivalent in $\mathcal{B}\pi$.

Exact equality is appropriate in finite symbolic settings. Approximate equivalence requires a certified tolerance and explicit uncertainty model.

Proposition 3.4. Monotonicity under policy refinement

If $\pi \leq \rho$, then

$$X \simeq_{\rho} Y \Rightarrow X \simeq_{\pi} Y.$$

Proof. A refined policy preserves every distinction visible under the coarser policy. Equality under ρ therefore implies equality under π . ■

Corollary 3.5

Policy refinement induces a canonical coarsening map

$$\mathcal{G}/\simeq_{\rho} \rightarrow \mathcal{G}/\simeq_{\pi}.$$

Thus policy-indexed object systems form a refinement structure.

4. Functional Quotient Formation

Definition 4.1. Policy quotient

The policy quotient $\mathcal{G}/\simeq_{\pi}$ is the category whose objects are policy-equivalence classes

$$[X]_{\pi} = \{Y \in \text{Ob}(\mathcal{G}) : Y \simeq_{\pi} X\},$$

with morphisms induced by admissible transformations compatible with $\simeq_{\pi}$.

The quotient functor is

$$Q\pi: \mathcal{G} \rightarrow \mathcal{G}/\simeq_{\pi}.$$

Definition 4.2. Neural object

A neural object under policy π is a class $[X]_\pi$ such that $\simeq_\pi$ is stable under the admissible contexts in which the class is used.

Thus a neural object is not merely a cluster of similar activations. It is a licensed substitution class.

Proposition 4.3. Contextual substitutability

Let $X \simeq_\pi Y$. Suppose each admissible context C defines a morphism preserved by $O\pi$. Then

$$O\pi(C(X)) \simeq O\pi(C(Y)).$$

Proof. Functoriality gives $O\pi(C(X)) = O\pi(C)(O\pi(X))$ and similarly for Y . Since $O\pi(X) \simeq O\pi(Y)$, compatibility of $O\pi(C)$ with behavioral equivalence preserves the relation. ■

Corollary 4.4

A policy-equivalence relation compatible with every admissible context acts as a neural congruence.

This is the direct analogue of congruence compatibility in universal algebra.

Proposition 4.5. Universal property

Let $F: \mathcal{G} \rightarrow \mathcal{D}$ be a functor such that $X \simeq_\pi Y$ implies $F(X) = F(Y)$. Then F factors through $Q\pi$:

$$F = \bar{F} \circ Q\pi$$

for a unique functor $\bar{F}: \mathcal{G}/\simeq_\pi \rightarrow \mathcal{D}$, up to the appropriate categorical equality.

This states that every policy-respecting computation may operate on quotient objects rather than their redundant presentations.

5. Conceptual Resolution

Different policies produce different internal ontologies.

A coarse policy may identify states that preserve only final task labels. A finer policy may distinguish states according to full probability distributions, robustness, causal effects, transfer behavior, or future computational role.

Definition 5.1. Conceptual resolution

The conceptual resolution associated with π is the quotient category $\mathcal{G}/\simeq_\pi$.

Definition 5.2. Object fission

An object $[X]\pi$ undergoes fission under refinement $\rho \geq \pi$ when its preimage in $\mathcal{G}/\simeq_\rho$ contains two or more distinct classes.

Definition 5.3. Object fusion

Two classes distinct under ρ fuse under π when they map to the same class under the coarsening map

$$\mathcal{G}/\simeq_\rho \rightarrow \mathcal{G}/\simeq_\pi.$$

These operations formalize changes in conceptual granularity. A concept is not assumed to possess one absolute resolution.

6. Training as Developmental Functoriality

Let $\mathcal{G}_t$ denote the generative category associated with training checkpoint t .

A training update induces a developmental transformation

$$U_t: \mathcal{G}_t \rightarrow \mathcal{G}_{t+1}.$$

The model develops through the sequence

$$\mathcal{G}_0 \rightarrow \mathcal{G}_1 \rightarrow \mathcal{G}_2 \rightarrow \dots$$

Definition 6.1. Continuation of an object

A quotient object $[X_t]\pi$ has a continuation at time $t+1$ when U_t maps all representatives of $[X_t]\pi$ into one well-defined class $[X_{t+1}]\pi$.

Definition 6.2. Developmental persistence

A quotient object is persistent over $[t_0, t_1]$ when its continuation is well-defined throughout the interval and its policy-relevant behavior remains stable.

Definition 6.3. Neural birth event

A neural birth event occurs at the earliest checkpoint or interval in which a quotient class satisfies:

stable continuation;

low within-class behavioral variation;

persistent between-class distinction;

causal substitutability;

recurrence across contexts;

survival under subsequent training or compression.

The first linear probe capable of predicting a label does not necessarily mark object birth. Probe decodability is correlational. Birth requires stable functional substitutability.

7. Compression as Quotient Functor

Let

$$C: \mathcal{G} \rightarrow \mathcal{G}'$$

represent pruning, quantization, factorization, bottlenecking, distillation, or another compression transformation.

Definition 7.1. Policy-faithful compression

Compression C is faithful under π when there exists an observation functor

$$O'\pi: \mathcal{G}' \rightarrow \mathcal{B}\pi$$

such that

$$O'\pi \circ C \simeq O\pi.$$

The behavioral diagram approximately commutes:

$$\begin{array}{ccc} \mathcal{G} & \xrightarrow{C} & \mathcal{G}' \\ O_\pi \downarrow & & \downarrow O'_\pi \\ \mathcal{B}_\pi & \xlongequal{\quad} & \mathcal{B}_\pi. \end{array}$$

Definition 7.2. Compression kernel

The compression kernel consists of distinctions erased by C while preserving π -observable behavior.

The kernel approximates the equivalence relation

$$X \sim_C Y \Leftrightarrow C(X) = C(Y).$$

Proposition 7.3

If C is policy-faithful, then distinctions in its kernel are irrelevant to every observable behavior encoded by $O\pi$.

The proposition is immediate from commutativity of the behavioral diagram.

Compression can play two distinct roles.

Compression reveals an existing object when the quotient relation was already behaviorally stable before compression.

Compression creates an object when capacity pressure causes previously distinct presentations to become substitutable.

These cases can be separated experimentally by comparing post-training compression with compression-constrained training.

8. Theory Peeling in Neural Models

Algebraic Abiogenesis removes consequences explained by a discovered law and analyzes the residual. An analogous process can be applied to neural computation.

Let B be a certified neural object or circuit explaining part of model behavior. Let $\text{Eff}(B)$ denote the set of causal effects attributable to B . Define the residual behavioral structure by removing or controlling for $\text{Eff}(B)$.

The residual is then searched for additional stable equivalence classes.

This process may reveal a layered developmental sequence:

dominant feature formation;

subtraction of its explained behavior;

discovery of residual circuits;

promotion of residual circuits into modules;

repetition at a higher level.

Theory peeling prevents the interpretability system from repeatedly rediscovering variations of one dominant feature.

9. Certified Memory

Ordinary memory stores presentations. Certified memory stores equivalence classes together with evidence licensing their use.

Definition 9.1. Certified memory object

A certified memory object is a tuple

$$M = ([X]\pi, \pi, C, R, \Omega),$$

where:

$[X]\pi$ is the functional class;

π is the governing policy;

C is its certificate;

R is a canonical or compressed representative;

Ω records known limitations and failure conditions.

The certificate C may include:

supporting contexts;

substitution trials;

adversarial tests;

counterexamples attempted;

compression survival;

cross-seed recurrence;

cross-architecture recurrence;

permitted compositions;

policy-refinement boundaries.

Definition 9.2. Certified memory morphism

A memory morphism $f: M \rightarrow N$ is a certified transformation between memory objects whose use preserves the policies governing M and N .

Definition 9.3. Memory admissibility

A stored object may be reused in context K only when its certificate entails that K belongs to the permitted context family.

Certified memory therefore stores not only a conclusion but its domain of lawful substitution.

This prevents uncontrolled transport of equivalences across incompatible policies.

10. Homotopy-Theoretic Interpretation

Policy equivalence admits a HoTT-style interpretation.

A quotient class may be treated as a type.

Its presentations are terms of the type.

A certificate of equivalence is a path between presentations.

Different certificates may themselves be related by higher paths.

One path may arise from direct substitution tests. Another may arise from compression. A third may arise from a sequence of module transformations. These witnesses need not be definitionally identical.

Higher coherence records whether independently obtained certificates compose consistently.

Definition 10.1. Policy path

A policy path $p: X =_{\pi} Y$ is a certified witness that X and Y are substitutable under π .

Definition 10.2. Policy-univalence principle

A certified policy equivalence between X and Y licenses their identification within the π -indexed universe.

This is not unrestricted global univalence. The identification is indexed by:

the observation policy;

the certificate standard;

the admissible contexts;

the tolerated deviation;

the domain of reuse.

Proposition 10.3. Policy-local transport

Given $p: X =_{\pi} Y$ and a policy-respecting construction P , there exists a transport operation

$\text{transport}_p: P(X) \rightarrow P(Y)$

preserving π -relevant behavior.

This is the type-theoretic form of contextual substitutability.

11. Higher Coherence

Pairwise equivalence is insufficient for a large memory system.

Suppose certificates exist for

$$X \simeq_{\pi} Y,$$

$$Y \simeq_{\pi} Z,$$

$$X \simeq_{\pi} Z.$$

The system must determine whether the direct certificate and the composite certificate agree up to an admissible higher equivalence.

Likewise, two compression paths

$$X \rightarrow X_1 \rightarrow R$$

and

$$X \rightarrow X_2 \rightarrow R'$$

may produce distinct representatives. Certified memory requires evidence that R and R' remain equivalent and that subsequent composition is path-independent within tolerance.

A higher categorical structure is therefore natural when:

multiple equivalence witnesses coexist;

objects are compressed by several routes;

modules are composed in different orders;

certificates are transported across policies;

memory objects are recursively promoted.

The role of higher coherence is practical. It prevents contradictions caused by locally valid but globally incompatible substitutions.

12. Recursive Object Promotion

A certified object may become a generator in the next developmental stage.

Let $[X]\pi$ and $[Y]\pi$ be certified classes. Their admissible composition generates a new presentation

$$f([X]\pi, [Y]\pi).$$

A new ecology of composite objects is formed. Further observational quotients are then computed.

The hierarchy is:

raw states;

functional classes;

features;

circuits;

modules;

algorithms;

theories of module composition.

This mirrors Circuit Abiogenesis:

NAND gates;

truth functions;

recurrent motifs;

half-adders and full-adders;

larger arithmetic systems.

It also mirrors Algebraic Abiogenesis:

anonymous operations;

term-function collisions;

stable identities;

quotient algebras;

free structures.

Recursive promotion is the mechanism by which one level of discovered objecthood becomes the substrate for the next.

13. No-Corpus Artificial Intelligence

A no-corpus system does not require a preexisting textual archive of named objects as its foundational ontology. It may begin with:

primitive signals or states;

anonymous transformations;

environmental interaction;

policy-indexed observation;

recursive generation;

counterexample search;

certified memory.

A developmental cycle is:

1. Generate candidate states, transformations, or traces.
2. Observe their behavior under one or more policies.
3. Detect provisional functional equivalences.
4. Search for adversarial contexts separating apparent equivalents.
5. Form quotients from the surviving relations.
6. Construct canonical or compressed representatives.
7. Store classes with certificates and limitations.
8. Promote certified objects to the generator set.
9. Repeat the process recursively.

The resulting system accumulates an ontology rather than merely accumulating a corpus.

Its memory contains structures whose identity was established through stable substitutability.

14. Minimal Experimental Architecture

A first implementation should use a small model trained on a compositional task. Suitable environments include:

finite-state transformations;

symbolic term normalization;

small arithmetic procedures;
graph rewriting;
cellular-automaton prediction;
anonymous circuit simulation.

Frequent checkpoints are required.

14.1. State acquisition

Collect internal states across many inputs, contexts, and training checkpoints.

14.2. Intervention structure

Define admissible interventions:

activation replacement;
feature ablation;
path patching;
subcircuit substitution;
state projection;
compressed-module replacement.

14.3. Policy family

Use policies of increasing strength:

π_1 : final answer preservation;
 π_2 : output-distribution preservation;
 π_3 : preservation under perturbation;
 π_4 : preservation across related tasks;
 π_5 : preservation of downstream causal effects.

14.4. Quotient induction

Construct classes from intervention-tested substitutability, not geometric similarity alone.

14.5. Developmental tracking

Track class birth, persistence, fission, fusion, migration, and extinction.

14.6. Compression tests

Apply pruning, quantization, low-rank reduction, bottlenecking, and distillation.

14.7. Certification

Retain only classes surviving adversarial substitution, compression, and cross-seed recurrence.

14.8. Promotion

Expose the surviving classes as explicit modules in a second training stage.

15. Falsifiable Predictions

Prediction 15.1

Functional substitutability will predict valid component replacement better than Euclidean or cosine similarity alone.

Prediction 15.2

Stable quotient classes will sharpen during training, exhibiting decreasing within-class causal variance.

Prediction 15.3

Some quotient classes will recur across random seeds and model widths despite representational misalignment.

Prediction 15.4

Compression will preserve stable quotient classes preferentially over unstable activation clusters.

Prediction 15.5

Objects surviving several independent compression methods will correspond to compact causal circuits.

Prediction 15.6

Explicit promotion of certified classes into modules will reduce model description length, training time, or sample complexity.

Prediction 15.7

Policy refinement will produce a structured refinement hierarchy rather than unrelated clusterings.

Prediction 15.8

Some probe-detectable features will fail causal substitution and therefore fail object certification.

Prediction 15.9

Compression-constrained training will induce earlier or sharper quotient formation than unconstrained training.

Prediction 15.10

Higher-order objects will appear when certified first-order objects are recursively composed and requotiented.

Failure of these predictions would weaken the Neural Abiogenesis Conjecture.

16. Neural Abiogenesis Conjecture

Conjecture 16.1. Neural Abiogenesis

For a sufficiently compositional task and a learning system under finite representational capacity, training generates policy-indexed functional equivalence relations on internal presentations. Some relations stabilize across checkpoints, contexts, models, and compression procedures. Their quotient classes support lawful substitution, causal composition, and recursive reuse and therefore function as internally constructed objects.

Conjecture 16.2. Strong Neural Abiogenesis

Intelligent abstraction consists substantially in the recursive generation, certification, compression, promotion, and composition of policy-indexed quotient objects.

17. Distinction Between Formal Result and Model Hypothesis

The categorical constructions above are mathematical once the relevant categories, functors, policies, and equivalence relations are specified.

The claim that contemporary neural networks actually organize their internal states according to these structures is empirical.

It has not been established that neural networks literally implement category theory or HoTT. Nor is it established that their hidden states form exact congruences or exact quotient categories.

The framework should therefore be interpreted at three levels.

The first level is formal: policy-respecting equivalence induces quotients and lawful substitution.

The second level is methodological: these constructions define measurable criteria for neural objecthood.

The third level is conjectural: training and compression may spontaneously produce such structures inside learned models.

The proposed experiments are designed to separate these levels.

18. Discussion

Category theory is well suited to artificial intelligence because it describes objects through transformations, composition, and universal behavior rather than intrinsic substance.

HoTT is well suited because it permits identity to carry witnesses, multiple equivalence paths, and higher coherence.

Algebraic Abiogenesis contributes the developmental mechanism. It explains how stable equivalence classes can arise from anonymous operations before a named mathematical object has been selected.

No-Corpus Artificial Intelligence extends the process into learning and memory. Instead of beginning from a fixed ontology inherited from language, the system generates candidate states, tests substitutability, forms quotients, stores certificates, and promotes the resulting objects.

The synthesis may be summarized as follows:

category theory describes how constructed objects compose;

HoTT describes their witnessed and coherent identities;

Algebraic Abiogenesis describes how stable identities arise;

Neural Abiogenesis describes their formation during training;

certified memory governs their retention and reuse.

19. Conclusion

Artificial intelligence need not be understood solely as a system that compresses a corpus of preexisting human objects.

It may also be understood as a system capable of constructing its own internal objects.

The primitive elements are presentations, transformations, contexts, and observations. Training generates an ecology of states and computations. Policies determine which differences matter. Functional equivalence identifies substitutable presentations. Quotient formation turns those

presentations into objects. Compression selects compact representatives. Certification records the conditions under which identity is valid. Recursive promotion makes the objects available as generators of higher structure.

The resulting developmental sequence is:

generation

→ observation

→ equivalence

→ quotient

→ compression

→ certification

→ memory

→ recursive reuse.

This sequence extends Mathematical It from Bit from abstract algebra into neural computation.

The central object is not a vector, neuron, token, or parameter.

It is a stable policy-indexed equivalence class of presentations whose differences may be forgotten without loss of admissible behavior.

References

1. Youvan, D. C. No-Corpus Artificial Intelligence from Algebraic Abiogenesis: Policy-Governed Object Formation, Theory Ecologies, and Certified Memory. July 2026. DOI: 10.13140/RG.2.2.12146.06088.
2. Youvan, D. C. Mathematical It from Bit: Algebraic Abiogenesis, Anonymous Finite Operations, and the Automated Birth of Abstract Algebra. July 2026. DOI: 10.13140/RG.2.2.25987.26403.
3. Youvan, D. C. From Algebraic Abiogenesis to Neural Abiogenesis: Model Compression, Functional Quotients, Recursive Circuit Formation, and the Ground-Up Birth of Mathematical Objects. July 2026. DOI: 10.13140/RG.2.2.15711.21927.
4. Youvan, D. C. How Mathematics Can Be Born: A Public Guide to Algebraic Abiogenesis and Mathematical It from Bit. July 2026. DOI: 10.13140/RG.2.2.30600.99847.
5. Youvan, D. C. Algebraic Abiogenesis. Source code and computational audits. GitHub repository: DougYouvan/algebraic-abiogenesis.
6. Mac Lane, S. Categories for the Working Mathematician. 2nd ed. Springer, 1998.
7. Awodey, S. Category Theory. 2nd ed. Oxford University Press, 2010.

8. Riehl, E. *Category Theory in Context*. Dover Publications, 2016.
9. The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study, 2013.
10. Voevodsky, V. *Univalent Foundations and the Equivalence Principle*. Foundational manuscripts and lectures.
11. Birkhoff, G. On the Structure of Abstract Algebras. *Proceedings of the Cambridge Philosophical Society* 31 (1935): 433–454.
12. Burris, S., and Sankappanavar, H. P. *A Course in Universal Algebra*. Springer, 1981.
13. Baez, J. C., and Stay, M. Physics, Topology, Logic and Computation: A Rosetta Stone. In *New Structures for Physics*. Springer, 2011.
14. Fong, B., and Spivak, D. I. *An Invitation to Applied Category Theory: Seven Sketches in Compositionality*. Cambridge University Press, 2019.
15. Fong, B., Spivak, D. I., and Tuyéras, R. Backprop as Functor: A Compositional Perspective on Supervised Learning. *Proceedings of the 34th Annual ACM/IEEE Symposium on Logic in Computer Science*, 2019.
16. Cruttwell, G. S. H., Gavranović, B., Ghani, N., Wilson, P., and Zanasi, F. *Categorical Foundations of Gradient-Based Learning*. European Symposium on Programming, 2022.
17. Tishby, N., Pereira, F. C., and Bialek, W. The Information Bottleneck Method. *Proceedings of the 37th Annual Allerton Conference on Communication, Control, and Computing*, 1999.
18. Achille, A., and Soatto, S. Emergence of Invariance and Disentanglement in Deep Representations. *Journal of Machine Learning Research* 19 (2018): 1–34.
19. Hinton, G., Vinyals, O., and Dean, J. Distilling the Knowledge in a Neural Network. 2015.
20. Han, S., Mao, H., and Dally, W. J. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. *International Conference on Learning Representations*, 2016.
21. Frankle, J., and Carbin, M. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. *International Conference on Learning Representations*, 2019.
22. Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of Neural Network Representations Revisited. *Proceedings of the 36th International Conference on Machine Learning*, 2019.

23. Geiger, A., Lu, H., Icard, T., and Potts, C. Causal Abstractions of Neural Networks. *Advances in Neural Information Processing Systems* 34, 2021.
24. Geiger, A., et al. Causal Abstraction: A Theoretical Foundation for Mechanistic Interpretability. *Journal of Machine Learning Research* 26, 2025.
25. Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., and Carter, S. Zoom In: An Introduction to Circuits. *Distill*, 2020.
26. Elhage, N., et al. Toy Models of Superposition. *Transformer Circuits Thread*, 2022.
27. Cunningham, H., Ewart, A., Riggs, L., Huben, R., and Sharkey, L. Sparse Autoencoders Find Highly Interpretable Features in Language Models. *International Conference on Learning Representations*, 2024.
28. Ferns, N., Panangaden, P., and Precup, D. Bisimulation Metrics for Continuous Markov Decision Processes. *SIAM Journal on Computing* 40 (2011): 1662–1714.
29. Zhang, A., McAllister, R., Calandra, R., Gal, Y., and Levine, S. Learning Invariant Representations for Reinforcement Learning Without Reconstruction. *International Conference on Learning Representations*, 2021.
30. Shannon, C. E. A Mathematical Theory of Communication. *Bell System Technical Journal* 27 (1948): 379–423, 623–656.

The author has published over 5,700 AI-assisted papers at:

<https://www.researchgate.net/profile/Douglas-Youvan/research> . Google Scholar has indexed these preprints, and if you want a particular reference, the AI mode of a Google search (Gemini) has efficiently catalogued these preprints.